**AMENDMENTS TO THE CLAIMS:**

This listing of claims will replace all prior versions, and listings, of claims in the application:

**Listing Of Claims:**

Please amend the claims as follows:

1.      (Currently Amended)  A method for presenting forms and publishing form data, the method comprising:

maintaining a field engine table, the field engine table comprising data identifying one or more fields of a form, the data identifying the one or more fields of the form comprising a name, a data type, and a version number for each of the one or more fields of the form;

receiving a request for a network resource including the form;

in response to the request, determining whether a previously compiled class file is to be utilized to respond to the request;

in response to determining that a previously compiled class file is not to be utilized to respond to the request, creating an executable class file utilizing the field engine table to retrieve: one or more field names, a the form name corresponding to each of the one or more field names of the form, the data type corresponding to each of the one or more field names of the form, and a the version number corresponding to each of the one or more field names of the form, wherein each of the one or more field names are identical to corresponding fields in the form, and each of the one or more field names are associated with a corresponding response data of the form, and each of the one or more field names are associated with a corresponding data type indicating a

-2-

type of input field to display for each of the one or more field names, wherein ~~a software component is not hard-coded with the one or more field names,~~ the executable class file ~~being configure~~ is configured to generate markup language for displaying the fields of the form in a web browser and wherein the executable class file is not hard coded with the one or more field names;

generating the markup language by executing the class file; ~~and~~

returning the markup language as a response to the request for a network resource; and

maintaining an output table for storing the response data, wherein maintaining the output table comprises:

receiving a request to publish the response data associated with each of the field names, and

storing the response data associated with each of the field names in the output table, the output table having output table fields with names identical to the field names, wherein storing the response data in the output table comprises writing the response data to output table fields matching the field names of the form such that the response data is written to output table fields with names identically matching the field names.

2. (Previously Presented) The method of Claim 1, wherein determining whether a previously compiled class file is to be utilized comprises determining whether the request for the network resource was a first request for the network resource.

3. (Previously Presented) The method of Claim 1, wherein determining whether a previously compiled class file is to be utilized comprises determining whether the request for the network resource was a first request for the network resource or whether a web server operative to provide the network resource was reset since the last time the network was accessed.

4.-7. (Canceled)

8. (Currently Amended) A computer system for presenting forms and publishing form data, the computer system comprising:

a field engine table comprising data identifying one or more fields to be utilized in a form, the data identifying the one or more fields to be utilized in the form comprising a name, a data type, and a version number for each of the one or more fields of the form;

a network resource including the form; and

a software component for receiving and responding to requests for the network resource, the software component operative to:

determine whether a previously compiled class file should be utilized to respond to a request for the network resource, to create an executable class file utilizing a field engine table to retrieve the one or more field names, a the form name corresponding to each of the one or more field names of the form, the data type corresponding to each of the one or more field names of the form, and the a version number corresponding to each of the one or more field names of the form, wherein each of the one or more field names are identical to corresponding

fields in the form, ~~and~~ <u>each of</u> the one or more field names are associated with a corresponding response data of the form, <u>and each of the one or more field names are associated with a corresponding data type indicating a type of input field to display for each of the one or more field names,</u> wherein a ~~software component is not hard-coded with the one or more field names,~~ the executable class file ~~being configure~~ <u>is configured</u> to generate markup language for displaying the fields of the form in a web browser <u>and wherein the executable class file is not hard coded with the one or more field names,</u> wherein a runtime extension is selected to create the executable class file based upon a file extension associated with the request,

in response to determining that a previously compiled class file should not be utilized, execute the class file, and respond to the request with the markup language generated by the execution of the class file, and

maintain an output table for storing the response data, wherein the software component being operative to maintain the output table comprises the software component operative to:

receive a request to publish response data associated with each of the field names, and

store the response data associated with each of the field names in the output table, the output table having output table field with names identical to the field names, wherein storing the response data in the output table comprises writing the response data to output table fields matching the field names of the form such that the response data is

written to output table fields with names identically matching the field names.

9.      (Previously Presented)  The computer system of Claim 8, wherein determining whether a previously compiled class file is to be utilized comprises determining whether the request for the network resource was a first request for the network resource or whether the software component was reset since a previous request for the network resource.

10.-13.      (Canceled)

14.      (Currently Amended)  A computer-readable medium comprising computer-readable instructions which, when executed by a computer, cause the computer to:

determine whether a request has been received for a network resource for providing a form;

in response to determining that a request for the network resource has been received, determine whether a previously compiled class file is to be utilized to respond to the request for the form;

in response to determining that a previously compiled class file is not to be utilized, create an executable class file utilizing a field engine table to retrieve one or more field names, a form name corresponding to each of the one or more field names of the form, a data type corresponding to each of the one or more field names of the form, and a version number corresponding to each of the one or more field names of the

form, wherein <u>each of</u> the one or more field names are identical to corresponding fields in the form<u>,</u> ~~and~~ <u>each of</u> the one or more field names are associated with a corresponding response data of the form, <u>and each of the one or more field names are associated with a corresponding data type indicating a type of input field to display for each of the one or more field names,</u> wherein ~~a software component is not hard-coded with the one or more field names,~~ the executable class file ~~being configure~~ <u>is configured</u> to generate markup language for displaying the fields of the form in a web browser <u>and the executable class file is not hard coded with the one or more field names, and</u> wherein a runtime extension is selected to create the executable class file based upon a file extension associated with the request; and

maintain an output table for storing the response data, wherein causing the computer to maintain the output table comprises causing the computer to:

receive a request to publish response data associated with each of the field names, and

store the response data associated with each of the field names in the output table, the output table having output table field with names identical to the field names, wherein storing the response data in the output table comprises writing the response data to output table fields matching the field names of the form such that the response data is written to output table fields with names identically matching the field names; and

execute the class file and transmitting the content generated by the class file in response to the request.

15.    (Previously Presented)  The computer-readable medium of Claim 14, further comprising computer-readable instructions which, when executed by a computer, cause the computer to:

execute the previously compiled class file in response to determining that the previously compiled class file is to be utilized and responding to the request with content generated by the previously compiled class file.

16.    (Previously Presented)  The computer-readable medium of Claim 15, further comprising computer-readable instructions which, when executed by a computer, cause the computer to utilize the previously compiled class file if the request for the network resource is not a first request for the network resource and if a software component for receiving the request has not been reset since a previous request for the network resource.

17.    (Canceled)

18.    (Previously Presented)  The method of Claim 1, further comprising:

receiving a submission of response data associate with the field names; and

saving the response data associated with the field names in an output table, wherein the output table has an identical name as the completed form and the response data for each field has an identical name as the field name of the field, whereby a software component does not have to be hard-coded with the field names.

19.     (Canceled)


20.     (Previously Presented)  The method of Claim 1, wherein creating an executable class file further comprises selecting a runtime extension based upon a file extension associated with the request.